ESIGELEC
ÉCOLE D'INGÉNIEURS-ES GÉNÉRALISTES

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

C Programming



Ángel Perles
angel.perles@upv.es

Contents

Where is C?
The price of C
Which C?
Style guides
Lecturer
Objectives (TBD)
Course plan
Assessment
Course materials (TBD)
Lab tools
Bibliografy
Let's start

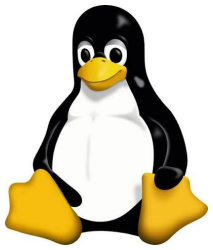
Where is C?

```
for (;;) // while(1)
{
```

- C (C++) language is “classic”, native, portable, modular, etc.
- The “universal” programming language for the embedded world and part of the rest
  - Any computer “toy” has assembler and C/C++ available
  - C/C++ is low/medium level --> ideal for peripheral programming
- Problems
  - low expressiveness, twisted, bugs-prone, ... hard for beginners, ...
- Why is all that stuff using C?








```
}
```

Do you disagree?: https://spectrum.ieee.org/top-programming-languages-2025
https://www.tiobe.com/tiobe-index/

The Price of C

- Ariane 5 Flight 501 self-destruction 37 seconds after launch
  - June 4, 1996. Kourou, French Guiana
  - Integer Overflow not trapped
  - Equivalent C code

```
double h_velocity = 40000.0;
int16_t h = (int16_t)h_velocity;   /* What happens in C? */
```

- MISRA/BARR-C would help here





Source: ESA

Which C?

- ISO C99
  - https://en.wikipedia.org/wiki/C99
  - Is the basis of standards such as MISRA-C:2012
  - It tries to minimise the issues of previous versions of C (e.g. C89 standard)
  - (But we will also keep under control the issues of this version!!)

```
$ gcc -std=c99 -Wall -Wextra -Wpedantic
```

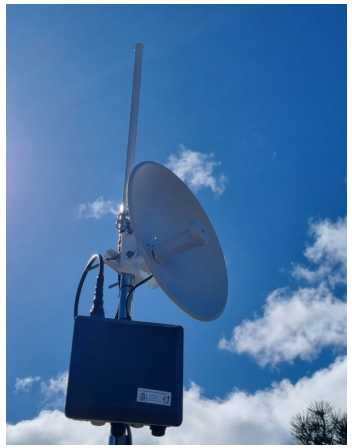
Style guides

- Style guides help fight bugs and share consistent code, among others
- E.g.:
  - MISRA-C is a well-recognised style guide from the automotive industry
  - SEI CERT C Coding Standard
- This course will follow *BARR-C:2018 Embedded C Coding Standard*
  - Aligned with MISRA-C:2012
  - https://barrgroup.com/embedded-c-coding-standard



Lecturer

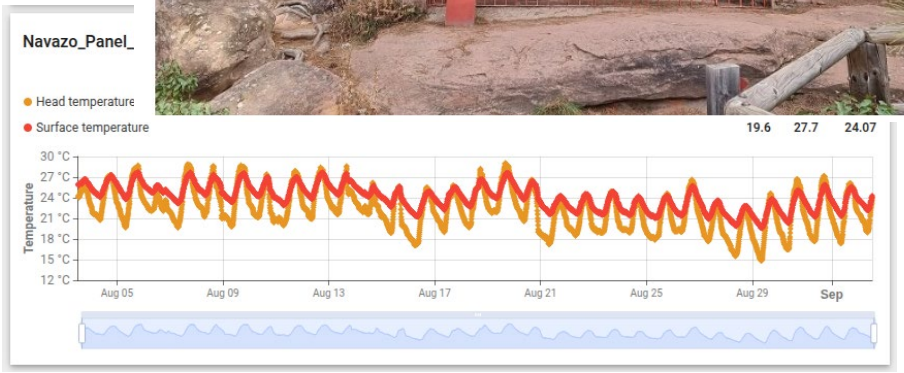
- Ángel Perles
  - Department of Computer Engineering
  - Universitat Politècnica de València (Valencia - Spain)
  - E-mail angel.perles@upv.es
  - Research focused on IoT -> full of C coding





UNIVERSITAT POLITÈCNICA DE VALÈNCIA



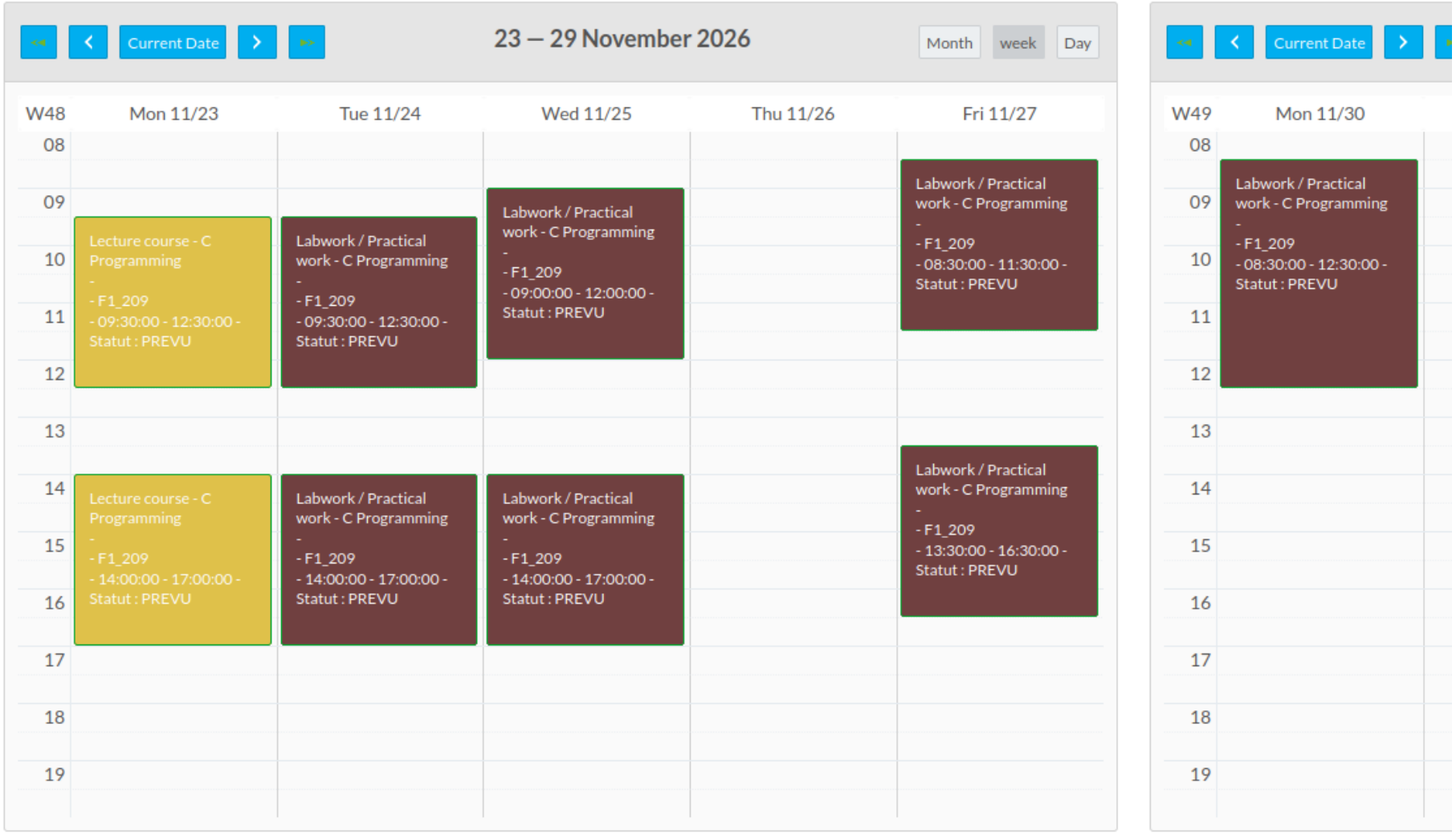


Objectives

- Build multi-file C programs
- Write clean, portable C99 code following the BARR-C style guide
- Use fixed-width integer types, *enum*, *struct* and *const* tables correctly
- Understand pointers and memory: arrays, output parameters, dynamic allocation
- Handle strings safely and parse text from devices and files
- Read and write text and binary files robustly
- Manipulate bits and registers with masks
- Spot the traps where C behaves differently from Java

Course plan

- Timetable



Course plan

- Building C projects
- Data types
- Pointers. Dynamic memory
- Control structures
- Strings. Printing and scanning
- Files
- Bitwise operations. Masks

All mixed with lab activities

And always tentative. I'll be flexible

Materials at: https://aperles.blogs.upv.es/esigelec-c-programming-mstee10/

Assessment

- Daily work (20%)
  - Show each exercise to the lecturer when you finish it
  - In groups of XX students
  - Submit it to XX in a zipped file
- Written exam (80%)
  - Monday 14 December
  - **Questions related to laboratory exercises**
  - One or two theoretical questions
  - Best practices coding

Course materials

- ????

Lab tools

- GNU gcc and associated toolset
- Microsoft Visual Studio Code





- I strongly suggest installing these tools in your laptops according the lab's guide
- Plan B: An online builder, e.g. https://godbolt.org/

Bibliografy

- K. N. King: C Programming: A Modern Approach, 2nd Edition, W. W. Norton, 2008. Covers C99, the standard used in this course.
- Michael Barr: Embedded C Coding Standard (BARR-C:2018), Barr Group, 2018. Freely available as PDF and HTML. All code submitted in the course should follow it. https://barrgroup.com/coding-standard
- Complementary reading
  - Robert C. Seacord: Effective C: An Introduction to Professional C Programming, 2nd Edition, No Starch Press, 2024. Focus on safe and secure C.
  - Jens Gustedt: Modern C, 3rd Edition, Manning, 2025.
- Historical reference
  - Brian W. Kernighan, Dennis M. Ritchie: The C Programming Language, 2nd Edition, Prentice Hall, 1988. The original reference by the creators of C; it describes ANSI C (C89) and predates current safe-coding practice.

Let's start …

- A C programming exam
  - A simple diagnostic of the group, do not worry
  - Not for marks



ESIGELEC

ÉCOLE D'INGÉNIEURS-ES GÉNÉRALISTES

UNIVERSITAT POLITÈCNICA DE VALÈNCIA